Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended): A method of error recovery of a bound remote method invocation (RMI) interface object, the method comprising:

binding an interface object of a parent process with ~~an~~ a first RMI process; and

starting a monitoring agent associated with the interface object, the monitoring agent comprising a thread to perform the steps of:

determining if ~~the interface object is bound with~~ the first RMI process is active and if the interface object is bound with any active RMI processes, wherein if the first RMI process is not active and if the interface object is not bound with any active RMI processes, an error occurs, the determining step including:

obtaining a bound uniform resource locator (URL) list from ~~the~~ active RMI processes; and

determining whether the interface object's name is in the bound URL list of the active RMI processes;

informing the interface object of the error when the monitoring agent determines that the interface object is not bound with any active RMI processes; and

rebinding the interface object with an active RMI process when the monitoring agent determines that ~~it's~~ the interface object is not bound with any active RMI processes, thereby recovering from the error without restarting the parent process that is bound with the first RMI process.

2-5. (cancelled).

6. (previously presented): The method of claim 1, comprising:

binding a second interface object of a second parent process with an RMI process; and

calling a second monitoring agent associated with the second interface object, the second monitoring agent comprising a second thread to perform the steps of:

monitoring the status of RMI processes; and

rebinding the second interface object with an active RMI process when the second monitoring agent determines that the second interface object is not bound with an active RMI process.

7. (previously presented): The method of claim 1, wherein the step of binding the interface object comprises:

binding one of an RMI daemon, a distributed task facility daemon, a log manager daemon, or a domain manager daemon, with an active RMI daemon.

8. (previously presented): The method of claim 1, comprising:

terminating the thread of the monitoring agent when the parent process is terminated.

9. (currently amended): A network system, comprising:

a plurality of remote nodes, at least one of the remote nodes running a remote method invocation (RMI) process; and

a management server for managing the remote nodes, the management server including at least one processor for running an RMI process and at least one management process that is bound with a first RMI process through an interface object, each at least one management process being associated with a monitoring agent comprising a thread to perform the steps of:

determining if ~~an interface object of a management process is bound with~~ the first RMI process is active and if the interface object is bound with any active RMI processes, wherein if the first RMI process is not active and if the interface object is not bound with any active RMI processes, an error occurs, the determining step including:

obtaining a bound uniform resource locator (URL) list from ~~the~~ active RMI processes; and

determining whether the interface object's name is in the bound URL list of the active RMI processes;

informing the interface object of the error when the monitoring agent determines that the interface object is not bound with any active RMI processes; and

rebinding the interface object with an active RMI process when the monitoring agent determines that ~~it's~~ the interface object is not bound with any active RMI

processes, thereby recovering from the error without restarting the parent process that is bound with the first RMI process.

10. (original): The network system of claim 9, wherein the at least one management process comprises a plurality of management processes.

11. (currently amended): The network system of claim 9, wherein the ~~plurality of~~ at least one management processes comprises:

a distributed task facility process;

a domain manager process; and

a log manager process.

12. (original): The network system of claim 9, wherein each of the remote nodes runs a service control manager agent process for performing server management tasks.

13. (original): The network system of claim 9, wherein the management server comprises:

a secondary storage device, the secondary storage device comprising:

a data repository;

a depot; and

a web server.

14. (original): The network system of claim 9, wherein the plurality of remote nodes are arranged into at least one node group, the network system comprising a service control manager for managing the at least one node group.

15. (currently amended): A method of error recovery of a bound remote method invocation (RMI) interface object, the method comprising:

a) performing a rebind call to ~~an~~ a first RMI process to provide a network address and an interface object of a parent process to the first RMI process; and

b) performing an initialization call to a monitoring agent associated with the interface object, the monitoring agent comprising a thread, the thread performing the steps of:

1) performing a list call to ~~an~~ active RMI processes to determine whether the interface object is bound with ~~the~~ any active RMI processes, the list call obtains a

bound uniform resource locator (URL) list from the active RMI processes and determines whether the parent process's name is in the bound URL list of the active RMI processes, wherein if the interface object is not bound with any active RMI processes, an error occurs;

2) informing the interface object of the error when the monitoring agent determines that the interface object is not bound with any active RMI processes;

3 ~~2~~) performing a rebind call to an active RMI process if the monitoring agent determines that the interface object is not bound with any active RMI processes, thereby recovering from the error without restarting the parent process that is bound with the first RMI process; and

4 ~~3~~) repeating steps 1 ~~and 2~~ through 3.

16. (original): The method of claim 15, wherein the parent process is one of a remote method invocation daemon, a distributed task facility daemon, a log manager daemon, and a domain manager daemon.

17-18. (cancelled).

19. (previously presented): The method of claim 15, comprising:
terminating the thread of the monitoring agent when the parent process is terminated.